

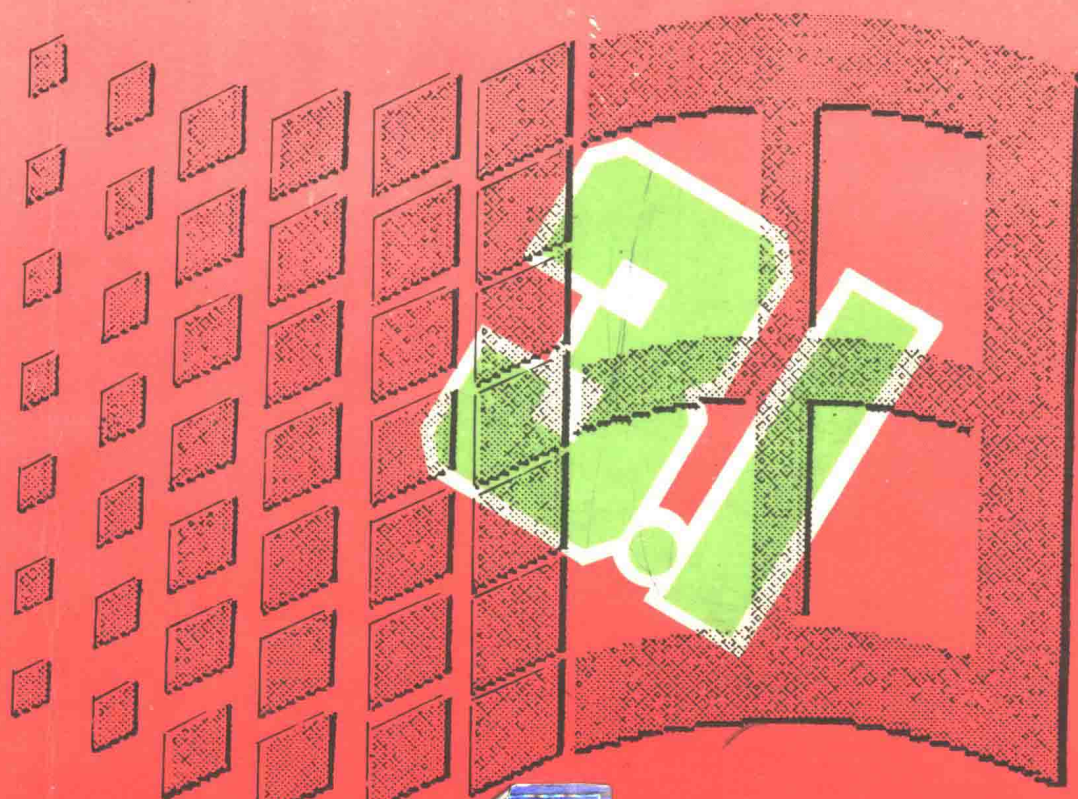
Microsoft Microsoft Microsoft

Microsoft Windows 3.1

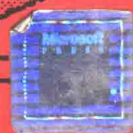
程序员参考大全(六)

—编程指南

吴之美 余立新 黄昱 译
毛大勇 沈金发



清华大学出版社



Microsoft

Windows 3.1

程序员参考大全(六) —编程指南

[美] Microsoft 公司 著
吴之美 余立新 黄 昱 译
毛大勇 沈金发

清 华 大 学 出 版 社

Microsoft Windows 3.1 Guide to Programming

本书英文版由 Microsoft 公司属下的 Microsoft 出版社(Microsoft Press)出版。
版权为 Microsoft 公司所有(Copyright ©1987—1992 by Microsoft Corporation)。
本书中文版经 Microsoft Press 授权清华大学出版社独家出版,1993。
未经出版者书面允许,不得用任何手段复制或抄袭本书内容。

Adobe®和 PostScript®是 Adobe Systems 公司的注册商标。

Apple®,Macintosh®和 True Type®是 AppleComputer 公司的注册商标。

PANOSE™是 ElseWare 公司的商标。

Epson®和 FX®是 Epson America 公司的注册商标。

Hewlett-Packard®,HP®,LaserJet®和 PCL®是 Hewlett-Packard 公司的注册商标。

IBM®是 International Business Machines 公司的注册商标。

ITC Zapf Chancery®和 ITC Zapf Dingbats®是 International Typeface 公司的注册商标。

Helvetica®,Palatino®,Times®和 Times Roman®是 Linotype AG 和(或)其下属的注册商标。

CodeView®,Microsoft®,MS®,MS-DOS®和 QuickC®是 Microsoft 公司的注册商标。

QuickBasic™和 Windows™是 Microsoft 公司的商标。

Arial®和 Times New Roman®是 Monotype Corporation PLC 的注册商标。

Okidata®是 Oki America 公司的注册商标。

(京)新登字 158 号

Microsoft Windows 3.1 程序员参考大全(六)——编程指南

[美] Microsoft 公司 著

吴之美 余立新 黄 昱 译

毛大勇 沈金发

☆

清华大学出版社出版

北京 清华园

清华大学印刷厂印刷

新华书店总店科技发行所发行

☆

开本:787×1092 1/16 印张:23.75 字数:555千字

1993年7月第1版 1993年7月第1次印刷

印数:0001—5000

ISBN 7-302-01225-3/TP·463

定价:39.00元

引 言

在使用本书之前需要了解一些背景知识,包括以下几个方面:

- Microsoft Windows 3.1 软件开发包(SDK)附带的手册
- 入门知识
- 本编程手册的目的和内容
- 为 Windows 操作系统编制应用程序所需的工具
- 本手册中所使用的范例程序
- 本手册中所使用的规则

软件开发包(SDK)手册集

该手册中,“SDK”特指 Microsoft Windows 3.1 软件开发包和它的内容。SDK 包括以下手册:

《Microsoft Windows 3.1 软件开发包(SDK)入门》提供了 SDK 的入门知识,解释了怎样安装 SDK 软件,并强调了 Windows 3.1 的变化。

《Microsoft Windows 编程指南》(本书)讲解了怎样编写 Windows 应用程序并提供了可用于开发实际应用程序的应用程序范例。本书还讲解了一些高级 Windows 编程技术。

《Microsoft Windows 编程工具》讲解怎样使用用于开发 Windows 应用程序的工具。这些工具包括调试器和特殊的 SDK 编辑器。

《Microsoft Windows 程序员参考大全》是一本内容广泛的编程手册,它提供了 Microsoft Windows 应用程序编程接口(API)所有的详细资料。该手册共分六册,按字母顺序列出了所有函数、宏、消息、数据类型和 API 数据结构,并提供了怎样使用 API 的广泛的指导。

入门知识

在学习本书之前,读者须具有以下条件:

- 使用过 Windows 系统,对 Windows 系统用户界面有基本了解。

在开发 Windows 应用程序之前,应先安装好 Windows 3.1 系统,并且学习如何使用它。一定要理解 Windows 应用程序各部分的名字、目的、操作方式(例如窗口、对话框、菜单、控制和滚动杠)。由于 Windows 应用程序肯定会使用它们,所以这些知识对 Windows 应用程序开发非常重要。

- 学习过 Windows 用户界面风格指南。Microsoft Windows 系统的一个目的是提供一致性的用户界面。统一的用户界面最终将减少应用程序用户学习时间,同时也

可以减少程序员设计用户界面的时间。

- 用 C 语言和 C 语言运行函数库编写过 C 语言程序。

Windows 应用程序推荐使用 C 语言编制程序(尽管 Windows 应用程序还可以用 PASCAL 和汇编语言编写,但这些语言在编制 Windows 应用程序时会遇到另外一些问题,而这些问题在 C 语言中则可以避免)。

关于本书

本书将帮助有经验的 C 语言程序员学习如何用 Windows 3.1 系统开发应用程序。本书讲解如何用 Windows 函数、消息和结构完成一些常用的任务,并通过一些程序范例演示这些技术的实际应用。本书所举范例均可以用 Windows 3.1 编译及运行。

本书共分三个部分,每一部分又分为几个章节。

第一部分,“在 Windows 环境中编程”,概述 Windows 操作系统;Windows 编程环境;并深入分析了一个 Windows 应用程序范例。第 1 部分包括以下几章:

- 第 1 章:“Windows 环境概述”。通过与 MS-DOS 编程环境比较,简单地介绍了 Windows 系统,讲解 Windows 应用程序各部分,列举 Windows 应用程序的开发过程。
- 第 2 章:“通用 Windows 应用程序”,详细讲述了一个简单的 Windows 应用程序——Generic——的开发。该应用程序也是本书所有程序范例的基础。

第二部分,“开发 Windows 应用程序”。讲解基本 Windows 应用程序开发技术,例如,创建菜单、打印和使用裁剪板。每章针对一个主题,并提供相关程序范例。分为以下几章:

- 第 3 章,“窗口输出”,介绍图形设备接口(GDI),讲述如何用 GDI 函数输出。
- 第 4 章,“键盘和鼠标输入”,介绍如何处理鼠标和键盘输入。
- 第 5 章,“图符”,介绍如何创建和显示图符。
- 第 6 章,“光标”,介绍如何创建和显示光标。
- 第 7 章,“菜单”,讲解如何创建菜单并处理菜单输入。
- 第 8 章,“控制”,介绍如何创建及使用控制,如按钮和列表框。
- 第 9 章,“对话框”,介绍如何创建和使用对话框,如何在对话框中使用控制。
- 第 10 章,“文件输入和输出”,讲解如何使用 OpenFile 函数,并给出从磁盘读写数据的规则。
- 第 11 章,“位图”,介绍如何创建和显示位图。
- 第 12 章,“打印”,介绍如何在 Windows 下使用打印机。
- 第 13 章,“裁剪板”,介绍裁剪板,并讲解如何使用裁剪板。

第三部分,“高级编程技术”。介绍并讲解一些高级编程技术,包括内存管理、动态数据交换(DDE)。每章针对一种高级技术。分为以下几章:

- 第 14 章,“C 和汇编语言”,介绍用 C 语言和汇编语言编写 Windows 应用程序的一些约定。
- 第 15 章,“内存管理”,讲解如何分配全局和局部内存。
- 第 16 章,“内存管理的进一步讨论”,深入分析如何高效使用内存。本章还讲述如何

• X •

在不同内存配置下使用内存。

- 第 17 章,“打印设置”,介绍如何配备打印机环境(如打印纸尺寸和方向)。
- 第 18 章,“字体”,介绍如何创建、装载和在 TextOut 函数中使用字体。
- 第 19 章,“彩色调色板”,讲解如何用彩色调色板充分利用色彩机制。
- 第 20 章,“动态连接库”,介绍如何创建及使用动态连接库(DLL)。
- 第 21 章,“多文本界面”,介绍如何开发使用多文本界面的应用程序,这种应用程序使用户可同时在几个文本中进行编程工作。
- 第 22 章,“动态数据交换”,讲解如何用基于消息的 DDE 协议在应用程序之间传递信息。

推荐工具

为开发 Windows 3.1 应用程序,需配备下述工具:

- Microsoft C 优化编译器(CL)
- Microsoft 分段可执行连接器(LINK)
- Microsoft Windows 资源编译器(RC)
- Microsoft 图像编辑器(IMAGEDIT.EXE)
- Microsoft 对话框编辑器(DLGEDIT.EXE)

Windows 库文件和字体文件需用下述工具创建:

- Microsoft 宏汇编(ML)
- Microsoft Windows 字体编辑器(FONTEDIT.EXE)

Windows 应用程序的制作和调试工具如下:

- Microsoft 程序维护机制(NMAKE)
- Microsoft Windows CodeView(CVW)
- Microsoft Windows 剖析器
- Microsoft Windows 堆观察器(HEAPWALK.EXE)
- Microsoft Windows 窥视器(SPY.EXE)

上述工具,除 CL, LINK, ML 和 NMAKE 外,在 SDK《Microsoft Windows 编程工具》一书中有较详细的介绍。

Windows 3.1 系统软件和硬件配置清单参见《Microsoft Windows 3.1 软件开发包(SDK)入门手册》一书。

程序范例

本书使用的程序范例用 C 语言编写,并遵循 Windows 应用程序用户界面风格约定。

符号约定

下述约定在本书中用于定义各种语法:

约 定	意 义
正体字或黑体字	表示要逐词逐字的写出。如资源定义语句或函数(MENU 或 CreateWindow)、命令或命令行选项(/nod),必须严格书写这些字符或文字。
斜体字	表示占位符或变量。须提供实际值。例如,语句 SetCursorPos(<i>x,y</i>)要替换 <i>x,y</i> 参数。
[]	括起选项。
	分隔“二选一”或“或”选择。
...	表示前面的项可重复。
BEGIN	
:	表示省略的代码段。
END	

此外,本书还使用一些文字规则以助于阅读:

规 则	意 义
“小号”大写字符	表示键名、键序列和组合键——如 ALT+SPACEBAR。
正常大写字符	指示文件名和路径、类型名、大多数结构名(也是黑体字)及常量。
单空格	分隔代码及语法。

目 录

引言	IX
----------	----

第一部分 在 Windows 环境中编程

第 1 章 Windows 环境概述 1

1.1 Windows 与 MS-DOS 的对比	3
1.1.1 用户界面	3
1.1.2 排队输入	4
1.1.3 设备无关图形	4
1.1.4 多任务机制	4
1.2 Windows 应用程序的元素	5
1.2.1 窗口	5
1.2.2 菜单	5
1.2.3 对话框	6
1.2.4 消息循环	6
1.3 Windows 库	8
1.4 软件开发工具	8
1.4.1 Microsoft C 优化编译器	9
1.4.2 Microsoft 分段可执行 连接器	9
1.4.3 资源编辑器	10
1.4.4 Microsoft Windows 资源 编译器	10
1.4.5 调试和优化工具	10
1.4.6 Microsoft 程序维护机制	11
1.5 开发 Windows 应用程序	12
1.6 有关内容	13

第 2 章 通用 Windows 应用程序 14

2.1 标准 Windows 应用程序: Generic	14
2.2 WinMain 函数	15
2.2.1 数据结构和数据类型	15
2.2.2 句柄	16
2.2.3 实例	16
2.2.4 登记一个窗口类	17

2.2.5 创建窗口	19
2.2.6 显示及更新窗口	20
2.2.7 编制消息循环	21
2.2.8 出让控制权	22
2.2.9 终止应用程序	22
2.2.10 初始化函数	23
2.2.11 应用程序命令行参数	24
2.3 窗口函数	24
2.4 创建 About 对话框	26
2.4.1 编制对话框模板	26
2.4.2 编制头文件	27
2.4.3 编制对话框函数	28
2.4.4 定义带有 About 命令的 菜单	29
2.4.5 处理 WM_COMMAND 消息	29
2.5 编制模块定义文件	30
2.6 创建 Generic	32
2.6.1 编制 C 语言源文件	32
2.6.2 编制头文件	37
2.6.3 编制资源定义文件	38
2.6.4 编制模块定义文件	38
2.6.5 编制 makefile 文件	39
2.6.6 运行 Microsoft 程序 维护机制	40
2.7 使用 Generic 作为模板	40
2.8 有关内容	41

第二部分 开发 Windows 应用程序

第 3 章 窗口输出 45

3.1 设备描述表的使用	45
3.1.1 使用 GetDC 函数	45
3.1.2 处理 WM_PAINT 消息	46
3.1.3 使用户域无效	47

3.1.4 准备设备描述表	47	Cases 语句	63
3.1.5 坐标系	47	4.2.12 增加 WM_LBUTTONDOWNBLCLK Case 语句	64
3.2 创建、选择及删除绘图工具	48	4.2.13 增加 WM_TIMER Case 语句	64
3.3 绘图和写字	49	4.2.14 增加 WM_HSCROLL 和 WM_VSCROLL Cases 语句	64
3.4 范例应用程序:Output	50	4.2.15 增加 WM_PAINT Case 语句	65
3.4.1 增加新变量	50	4.2.16 编译和连接	65
3.4.2 增加 WM_CREATE Case 语句	51	4.3 有关内容	66
3.4.3 增加 WM_PAINT Case 语句	51	第 5 章 图符	67
3.4.4 修改 WM_DESTROY Case 语句	54	5.1 什么是图符	67
3.4.5 编译和连接	55	5.2 使用系统提供的图符	67
3.5 有关内容	55	5.3 使用自定义的图符	68
第 4 章 键盘和鼠标输入	56	5.3.1 创建图符文件	68
4.1 Windows 输入消息	56	5.3.2 定义图符资源	68
4.1.1 消息格式	56	5.3.3 装载图符资源	69
4.1.2 键盘输入	57	5.4 指定类图符	69
4.1.3 字符输入	57	5.5 显示自定义的图符	69
4.1.4 鼠标输入	57	5.6 在对话框中显示图符	71
4.1.5 计时器输入	58	5.7 应用程序范例:Icon	71
4.1.6 滚动杠输入	59	5.7.1 增加 ICON 语句	72
4.1.7 菜单输入	59	5.7.2 增加 ICON 控制语句	72
4.2 范例应用程序:Input	60	5.7.3 设置类图符	72
4.2.1 Input 应用程序如何显示 输出信息	60	5.7.4 在 makefile 文件中增加 MYICON.ICO 行	72
4.2.2 增加新变量	61	5.7.5 编译和连接	72
4.2.3 设置窗口类风格	61	5.8 有关内容	72
4.2.4 修改 CreateWindow 函数	61	第 6 章 光标	73
4.2.5 设置文字矩形	62	6.1 控制光标形状	73
4.2.6 增加 WM_CREATE Case 语句	62	6.1.1 使用系统光标形状	73
4.2.7 修改 WM_DESTROY Case 语句	63	6.1.2 使用自定义光标形状	74
4.2.8 增加 WM_KEYUP 和 WM_ KEYDOWN Cases 语句	63	6.2 显示光标	74
4.2.9 增加 WM_CHAR Case 语句	63	6.2.1 指定类光标	74
4.2.10 增加 WM_MOUSEMOVE Case 语句	63	6.2.2 直接设置光标形状	75
4.2.11 增加 WM_LBUTTONUP 和 WM_LBUTTONDOWN		6.2.3 范例:在长时间的处理中 显示时间瓶光标	75
		6.3 用鼠标使用光标	76
		6.3.1 开始图形选择	76

6.3.2 扩充图形选择	78	7.6 使用特殊菜单特性	101
6.3.3 显示图形选择	78	7.6.1 为菜单和菜单项提供 加速键	101
6.3.4 结束图形选择	79	7.6.2 使用级联式菜单	103
6.4 用键盘使用光标	80	7.6.3 使用浮动弹出菜单	104
6.4.1 移动光标	80	7.6.4 设计自己的检取标志	106
6.4.2 当无鼠标时使用光标	82	7.6.5 使用自画菜单项	106
6.5 范例应用程序:Cursor	83	7.7 应用程序示例:EditMenu	107
6.5.1 增加 CURSOR 语句	83	7.7.1 在资源定义文件中增加 新菜单	108
6.5.2 增加新变量	84	7.7.2 在头文件中增加定义	109
6.5.3 设置类光标	84	7.7.3 在资源定义文件中增加 加速键表	109
6.5.4 准备时间瓶光标	84	7.7.4 增加新变量	110
6.5.5 增加一个长操作	85	7.7.5 装载加速键表	110
6.5.6 增加 WM_LBUTTONDOWN, WM_MOUSEMOVE 和 WM_ LBUTTONUP Case 语句	85	7.7.6 修改消息循环	110
6.5.7 增加 WM_KEYDOWN 和 WM_ KEYUP Cases 语句	87	7.7.7 修改 WM_COMMAND Case 语句	110
6.5.8 增加 WM_PAINT Case 语句	89	7.7.8 编译和连接	111
6.5.9 在 makefile 文件中增加 BULLSEYE.CUR 行	89	7.8 有关内容	111
6.5.10 编译和连接	89	第 8 章 控制	112
6.6 有关内容	90	8.1 什么叫控制	112
第 7 章 菜单	91	8.2 创建控制	112
7.1 菜单与菜单项	91	8.2.1 指定控制类	113
7.2 定义菜单	91	8.2.2 选择控制风格	113
7.2.1 菜单标识符	92	8.2.3 设置父窗口	114
7.3 在应用程序中包含菜单	93	8.2.4 选择控制标识符	114
7.3.1 指定一个窗口类的菜单	93	8.3 使用控制	114
7.3.2 指定一个特定窗口的菜单	93	8.3.1 接收用户输入	114
7.4 处理菜单输入	94	8.3.2 发送控制消息	115
7.5 从应用程序中控制菜单	95	8.3.3 使对一个控制的输入无效 或有效	115
7.5.1 使菜单项有效或无效	95	8.3.4 移动和改变控制的大小	115
7.5.2 检取和清除菜单项	96	8.3.5 销毁控制	116
7.5.3 增加菜单项	96	8.4 创建和使用一些通用控制	116
7.5.4 修改已有菜单	97	8.4.1 按钮	116
7.5.5 删除菜单项	98	8.4.2 静态控制	119
7.5.6 用位图作为菜单项	98	8.4.3 列表框	120
7.5.7 替换菜单	99	8.4.4 组合框	123
7.5.8 创建新菜单	99	8.4.5 编辑控制	124
7.5.9 初始化菜单	100	8.4.6 滚动杠	125
		8.5 应用程序示例:EditCntl	126

8.5.1 在头文件中增加一个常量 ...	126	10.9.8 增加 SaveAsDlg 对话框 函数	143
8.5.2 增加新变量	127	10.9.9 增加 Helper 函数	145
8.5.3 增加 CreateWindow 函数	127	10.9.10 输出 SaveAsDlg 对话框 函数	147
8.5.4 修改 WM_COMMAND Case 语句	128	10.9.11 增加堆空间	148
8.5.5 增加 WM_SETFOCUS Case 语句	128	10.9.12 编译和连接	148
8.5.6 增加 WM_SIZE Case 语句	128	10.10 有关内容	148
8.5.7 编译和连接	128	第 11 章 位图	149
8.6 有关内容	129	11.1 什么叫位图	149
第 9 章 对话框	130	11.2 创建位图	149
9.1 什么叫对话框	130	11.2.1 创建和装载一个位图 文件	150
9.1.1 模式对话框	130	11.2.2 创建并填充一个空白 位图	150
9.1.2 无模式对话框	131	11.2.3 用硬编码位数据创建一个 位图	151
9.2 使用对话框	131	11.2.4 绘制一个彩色位图	154
9.2.1 创建对话框过程	132	11.3 显示一个位图	155
9.2.2 在对话框中使用控制	133	11.3.1 用 BitBlt 函数显示一个 内存位图	155
9.3 有关内容	133	11.3.2 拉伸一个位图	157
第 10 章 文件输入和输出	134	11.3.3 在样式画刷中使用位图 ...	157
10.1 概述	134	11.3.4 显示一个设备无关位图 ...	158
10.2 Windows 系统中处理文件的 规则	134	11.4 在单色位图中增加颜色	159
10.3 创建文件	136	11.5 删除一个位图	160
10.4 打开一个存在文件	136	11.6 范例应用程序:Bitmap	160
10.5 读写文件	137	11.6.1 修改头文件	161
10.6 重新打开一个文件	137	11.6.2 增加位图资源	161
10.7 提请文件	138	11.6.3 增加 Bitmap, Pattern 和 Mode 菜单	162
10.8 检查打开文件的状态	138	11.6.4 增加全局和局部变量	162
10.9 范例应用程序:EditFile	138	11.6.5 增加 WM_CREATE Case 语句	163
10.9.1 增加头文件中的定义	138	11.6.6 修改 WM_DESTROY Case 语句	166
10.9.2 增加 SaveAs 对话框	138	11.6.7 增加 WM_LBUTTONDOWN, WM_MOUSEMOVE 和 WM_LBUTTONDOWN Cases 语句	166
10.9.3 增加 Include 语句	139	11.6.8 增加 WM_RBUTTONDOWN Case 语句	167
10.9.4 增加新变量	139		
10.9.5 修改 WM_COMMAND Case 语句	140		
10.9.6 增加 WM_QUERYEND- SESSION 和 WM_CLOSE Cases 语句	142		
10.9.7 修改 OpenDlg 对话框 函数	143		

11.6.9 增加 WM_ERASEBKGD	13.1.1 拷贝文字到裁剪板	190
Case 语句	13.1.2 从裁剪板粘贴文字	192
11.6.10 修改 WM_COMMAND	13.1.3 从裁剪板粘贴位图	194
Case 语句	13.1.4 Windows 裁剪板应用	195
11.6.11 修改 Makefile 文件	程序	195
11.6.12 编译和连接	13.2 使用特殊裁剪板特性	195
11.7 有关内容	13.2.1 在请求时补偿数据	195
第 12 章 打印	13.2.2 在终止前补偿格式	195
12.1 在 Windows 中打印	13.2.3 登记私有格式	196
12.1.1 函数和打印机指令	13.2.4 控制裁剪板中的数据	196
12.2 检取当前打印机的信息	显示	196
12.3 打印一行文字	13.3 范例应用程序:ClipText	198
12.4 打印一个位图	13.3.1 增加新变量	198
12.5 在打印过程中处理错误	13.3.2 修改实例初始化代码	199
12.6 取消打印操作	13.3.3 增加 WM_INITMENU Case	199
12.6.1 定义取消打印操作的	语句	199
对话框	13.3.4 修改 WM_COMMAND Case	200
12.6.2 定义 Abort 对话框的	语句	200
对话框函数	13.3.5 增加 WM_PAINT Case	201
12.6.3 定义取消打印操作的	语句	201
函数	13.3.6 增加 OutOfMemory	202
12.6.4 执行一个可取消的	函数	202
打印操作	13.3.7 编译和连接	202
12.6.5 用 AbortDoc 函数取消	13.4 有关内容	203
一个打印操作		
12.7 使用分段技术打印图像		
12.8 范例应用程序:PrintFile	第三部分 高级编程技术	
12.8.1 增加 AbortDlg 对话框	第 14 章 C 和汇编语言	207
12.8.2 增加打印用变量	14.1 选择内存模式	207
12.8.3 增中 IDM_PRINT Case	14.2 使用 NULL 常数	208
语句	14.3 使用命令行参数和 MS-DOS	208
12.8.4 编制 AbortDlg 对话框函数和	环境	208
AbortProc 函数	14.4 编写输出函数	209
12.8.5 增加 GetPrinterDC	14.4.1 创建回调函数	209
函数	14.4.2 创建 WinMain 函数	210
12.8.6 输出 AbortDlg 对话框函数和	14.5 使用 C 运行时间函数	210
AbortProc 函数	14.5.1 使用 Windows 的	210
12.8.7 编译和连接	C 程序库	210
12.9 有关内容	14.5.2 内存分配	211
第 13 章 裁剪板	14.5.3 处理字符串	211
13.1 使用裁剪板	14.5.4 文件输入和输出	212
	14.5.5 控制台输入和输出	213

14.5.6 图形函数	213	16.6.1 代码段属性	250
14.5.7 浮点运算	213	16.6.2 多代码段	250
14.5.8 运行其它的应用程序	213	16.6.3 代码段的平衡	250
14.5.9 BIOS 和 MS-DOS 接口 函数	214	16.7 有关内容	251
14.5.10 删除 C 运行启动码	214	第 17 章 打印设置	252
14.6 编制汇编语言代码	215	17.1 概述	252
14.6.1 改变中断标志	216	17.2 Windows 如何管理打印设置	253
14.6.2 用汇编语言写输出函数	217	17.2.1 打印设置和 DEVMODE 结构	253
14.6.3 ES 寄存器	218	17.2.2 打印设置和打印机环境	254
14.7 有关内容	219	17.3 使用设备驱动器函数	254
第 15 章 内存管理	220	17.4 确定打印机驱动器的性能	256
15.1 内存	220	17.5 使用打印设置	256
15.1.1 全局堆	221	17.5.1 确定 ExtDeviceMode 的 输入和输出	256
15.1.2 局部堆	221	17.5.2 检取打印设置的副本	257
15.1.3 可删除内存	223	17.5.3 修改打印设置	258
15.2 段	224	17.5.4 用 CreateDC 函数修改打印 设置	258
15.2.1 代码段	224	17.5.5 在不影响其它应用程序的 前提下修改打印设置	260
15.2.2 DATA 语句	225	17.5.6 提示用户修改打印设置	260
15.3 范例应用程序:Memory	225	17.6 在驱动程序间拷贝打印设置	261
15.3.1 分离 C 语言源文件	226	17.7 维护自定义的打印设置	262
15.3.2 修改头文件	226	17.8 在旧版本的打印驱动程序下 工作	262
15.3.3 增加新段定义	226	17.9 有关内容	263
15.3.4 修改 makefile 文件	227	第 18 章 字体	264
15.3.5 编译和连接	228	18.1 字体原理	264
15.4 有关内容	228	18.1.1 字体的组织原理	264
第 16 章 内存管理的进一步讨论	229	18.1.2 测量字符	265
16.1 内存结构	229	18.1.3 测量行距和字符内空间	266
16.1.1 标准模式	229	18.1.4 字符集	266
16.1.2 386 增强模式	232	18.2 Windows 中的字体	268
16.2 数据存储	233	18.2.1 光栅点阵(Raster)、矢量 和 TrueType 字体	268
16.2.1 自动数据段管理	234	18.2.2 字体资源文件	269
16.2.2 局部动态数据对象管理	236	18.2.3 TrueType 字体基础	270
16.2.3 全局内存对象管理	239	18.2.4 文字和字符属性	272
16.2.4 窗口和类结构的额外 字节	243	18.2.5 字体映射器	275
16.2.5 资源管理	244	18.2.6 标准字体对话框	275
16.3 内存模式	246		
16.4 巨型数据	247		
16.5 管理程序数据时要避免的陷阱	248		
16.6 程序代码的内存管理	249		

18.3	TrueType 字体技术	276	19.6.2	响应 WM_PALETTECHANGED 消息	311
18.3.1	所见即所得; WYSIWYG	276	19.7	有关内容	312
18.3.2	嵌入字体	277	第 20 章 动态连接库	313	
18.3.3	打印机可移植性	279	20.1	什么是动态连接库	313
18.3.4	文本的可移植性	280	20.1.1	输入库和动态连接库	314
18.3.5	磁盘空间、内存用法和速度	280	20.1.2	应用程序模块和动态连接模块	314
18.3.6	字体设计和比例变换	280	20.1.3	动态连接库和任务	315
18.3.7	设计可移植性字体	281	20.1.4	动态连接库和栈	315
18.4	在应用程序中使用字体	282	20.1.5	Windows 如何定位动态连接库	316
18.4.1	使用库存字体	282	20.2	什么时候使用自定义的动态连接库	316
18.4.2	列举字体	283	20.2.1	在应用程序间共享对象	316
18.4.3	检查设备的文字能力	285	20.2.2	为不同市场移植应用程序	317
18.4.4	创建逻辑字体	286	20.2.3	Windows 钩子函数	318
18.4.5	检取关于所选字体的信息	287	20.2.4	设备驱动器	318
18.4.6	检取关于逻辑字体的信息	288	20.2.5	自定义控制	319
18.4.7	绘制文字	288	20.2.6	项目管理	323
18.4.8	True Type 字体函数和结构	292	20.3	创建动态连接库	323
18.4.9	创建自定义字体	296	20.3.1	创建 C 语言源文件	324
18.5	有关内容	299	20.3.2	编制模块定义文件	329
第 19 章 彩色调色板	300	20.3.3	编制 Makefile 文件	330	
19.1	彩色调色板做什么	300	20.4	应用程序访问动态连接代码	332
19.2	彩色调色板如何工作	300	20.4.1	创建库函数的原型	332
19.3	创建和使用逻辑调色板	302	20.4.2	输入库函数	332
19.3.1	创建 LOGPALETTE 结构	302	20.5	Windows 对象所有权规则	334
19.3.2	创建逻辑调色板	305	20.6	范例动态连接库: Select	334
19.3.3	将调色板选入设备描述表	305	20.6.1	编制函数	335
19.3.4	实现调色板	305	20.6.2	编制初始化函数	339
19.4	用调色板的颜色绘图	305	20.6.3	编制退出函数	339
19.4.1	直接指定调色板颜色	305	20.6.4	编制模块定义文件	339
19.4.2	间接指定调色板颜色	306	20.6.5	编制头文件	340
19.4.3	画位图时使用调色板	307	20.6.6	编译和连接	340
19.5	修改逻辑调色板	308	20.7	有关内容	340
19.6	响应系统调色板的修改	310	第 21 章 多文本界面	341	
19.6.1	响应 WM_QUERYWPALETTE_MESSAGE 消息	310	21.1	多文本界面应用程序的成分	341
			21.2	初始化多文本界面应用程序	342
			21.2.1	登记窗口类	342

21.2.2 创建窗口	342	用处	349
21.3 编写主消息循环	343	22.1.5 从用户角度看到的动态 数据交换	350
21.4 编写边框窗口函数	344	22.2 动态数据交换的概念	350
21.5 编写子窗口函数	344	22.2.1 客户、服务器和会话	350
21.6 将数据与子窗口相联	345	22.2.2 应用程序名、主题名和 项名	351
21.6.1 在窗口结构中存放数据	345	22.2.3 永久数据连接	351
21.6.2 使用窗口特征机制	345	22.2.4 原子和共享内存对象	351
21.7 控制子窗口	345	22.3 动态数据交换消息	352
21.7.1 创建子窗口	346	22.4 动态数据交换消息流程	352
21.7.2 删除子窗口	346	22.4.1 启动一个会话	353
21.7.3 激活或挂起一个子窗口	346	22.4.2 发送一个数据项	355
21.7.4 在屏幕上安排子窗口	347	22.4.3 建立永久数据连接	359
21.8 有关内容	347	22.4.4 执行远程应用程序中的 命令	364
第 22 章 动态数据交换	348	22.4.5 终止会话	364
22.1 Windows 中的数据交换	348	22.5 System 主题	365
22.1.1 裁剪板传送	349	22.6 有关内容	365
22.1.2 动态连接库	349		
22.1.3 动态数据交换	349		
22.1.4 Windows 动态数据交换的			

第一部分

在 Windows 环境中编程

